REMARKS

Claims 1-5, 7-10, 12-16, 18-21, and 23-25 are pending in this application.

Applicant has amended claims 1, 7, 10, 18, 21, and 23, and has canceled claims 6, 17, and 22. In addition, Applicant has made a minor change to the specification to remove an embedded hyperlink. These changes do not introduce any new matter.

Objection to the Specification

In response to the objection to the specification, Applicant has replaced the embedded hyperlink with "<java.sun.com/products/javacard>," as suggested by the Examiner. Accordingly, Applicant requests that the objection to the specification be withdrawn.

Claim Amendments

Applicant has amended independent method claim 1 to recite "the first and the second virtual machine both access a common heap in a non-volatile memory of the data carrier, wherein write operations to the common heap are only performed by one of the first and second virtual machines." The feature that the first and second virtual machines both access a common heap is recited in original claim 6, which has been canceled herein. Support for the feature that write operations to the common heap are only performed by one of the first and second virtual machines can be found in Paragraph [0017] of the English translation of the as-filed PCT application.

Applicant has amended independent claim 10, which defines a portable data carrier, and independent claim 21, which defines a computer program product, along the same lines that claim 1 has been amended. In addition, claim 10 has been further amended to specify that the portable data carrier includes a non-volatile memory. Claim 21 has been further amended to specify that the computer program product is a "tangible" computer program product.

Rejection Under 35 U.S.C. § 112

Applicant respectfully requests reconsideration of the rejection of claims 9, 20, and 25 under 35 U.S.C. § 112, second paragraph, as being indefinite. In support of the rejection, the Examiner alleges that claims 9, 20, and 25 are indefinite because they contain the trademark/trade name "Java Card." Applicant respectfully traverses the indefiniteness rejection.

Section 608.01(v) of the MPEP states, in part, as follows:

> The expressions "trademarks" and "names used in trade" as used below have the following meanings:
>
> *Trademark:* a word, letter, symbol, or device adopted by one manufacturer or merchant and used to identify and distinguish his or her product from those of others. It is a proprietary word, letter, symbol, or device pointing distinctly to the product of one producer.
>
> *Names Used in Trade:* a nonproprietary name by which an article or product is known and called among traders or workers in the art, although it may not be so known by the public, generally. Names used in trade do not point to the product of one producer, but they identify a single article or product irrespective of producer.
>
> Names used in trade are permissible in patent applications if:
>
> (A) Their meanings are established by an accompanying definition which is sufficiently precise and definite to be made part of the claim, or
>
> (B) In this country, their meanings are well-known and satisfactorily defined in the literature.

Applicant respectfully submits that the term "Java Card," even if it contains the trademarked word "Java," should be regarded as a "name used in trade" according to the above definitions. This is because the term "Java Card" denotes a certain technology that is implemented in smart cards of various producers. In this regard, Applicant notes that a total of 29 companies have licensed the Java Card technology and ship corresponding products (see the Web page at --http://java.sun.com/javacard/licensees/--). Thus, the term "Java Card" is a name used in trade to denote any smart card, irrespective of its producer, that contains Java Card technology.

Furthermore, the meaning of the term "Java Card" is well known and defined in the literature. In fact, the Java Card technology is fully documented in specifications that are publicly (and freely) available at --http://java.sun.com/javacard/--. Thus, the Java Card technology is much better defined than many other standardized technologies, for which the official standard documents are available only at high costs or under a non-disclosure agreement.

Yet further, Applicant notes that the Office routinely grants patents that have claims that contain the terms "Java" and "Java Card." Recent searches in the Office's PatFT database revealed over 2,400 U.S. patents having claims that contain the registered trademark "Java," as well as 13 U.S. patents having claims contain either the term "Java Card" or "JavaCard."

In view of the foregoing, Applicant submits that the use of the term "Java Card" in claims 9, 20, and 25 is in compliance with MPEP § 608.01(v) and does not render the scope of the claimed subject matter indefinite. Accordingly, Applicant requests that the rejection of claims 9, 20, and 25 under 35 U.S.C. § 112, second paragraph, be withdrawn.

Rejection Under 35 U.S.C. § 101

Applicant respectfully requests reconsideration of the rejection of claims 21-25 under 35 U.S.C. § 101 as being directed toward non-statutory subject matter. In response to the Examiner's concerns regarding non-statutory subject matter, Applicant has amended independent claim 21 to define a "tangible" computer program product. Accordingly, Applicant submits that claims 21-25 now define statutory subject matter under 35 U.S.C. § 101, and requests that the rejection of these claims thereunder be withdrawn.

Rejection Under 35 U.S.C. § 103

Applicant respectfully requests reconsideration of the rejection of claims 1-10 and 12-25 under 35 U.S.C. § 103(a) as being unpatentable over *Osen* (EP 1 271 317 A1) in view of

Starovic et al. ("*Starovic*") (US 6,625,751 B1). As will be explained in more detail below, the combination of the *Osen* and *Starovic* references would not have suggested to one having ordinary skill in the art the subject matter defined in independent claims 1, 10, and 21, as amended herein.

Independent Claims 1, 10, and 21

In support of the obviousness rejection, the Examiner asserts that it would have been obvious to one having ordinary skill in the art to combine the smart card operating system of *Osen* with the virtual machines of *Starovic*.

In response, Applicant notes that the System-on-Chip of *Osen* is a small electronic module in which all elements (such as a processor, a memory, and an input/output interface) are integrated within a single semiconductor chip. The software architecture is just as simplified. In fact, *Osen* does not even distinguish between operating system and application software. Instead, all executed code is called the "operating system." Paragraph [0003] of *Osen* states:

> In a System-on-Chip the applications are often indistinguishable from the operating system kernel, and in the context of this patent **the applications (code and data) are considered to be part of the operating system.** [Emphasis added.]

On the other hand, *Starovic* discloses a sophisticated system architecture with multiple processing nodes 12, 14 that are linked by a network connection 16 and optionally via an additional connection 18. See Figure 1 and column 9, lines 40-41. Each node 12, 14 is a full-fledged computer. See Figure 2 and column 9, lines 53-54. Furthermore, in each node 12, 14 there is a clear separation between an operating system 50, a virtual machine 52, and one or more applications 54. Seee Figure 3 and column 10, lines 4-9.

The *Osen* and *Starovic* references represent the opposite extremes in computing technology. It would not have been obvious for the ordinarily skilled person to implement any features that may be known from multi-node computing networks in a single-chip system.

As discussed above, Applicant has amended claims 1, 10, and 21 to incorporate, among other features, the feature that the first and the second virtual machine both access a common heap in a non-volatile memory of the data carrier. This feature corresponds to original claim 6. In support of the obviousness rejection, the Examiner indicated that this feature was known based on the description at column 3, lines 20-37 of *Osen*.

In response, Applicant notes that the obviousness rejection depends on the combination of the *Osen* and *Starovic* references; however, the *Starovic* reference expressly teaches away from the above-mentioned feature. At column 5, lines 47-51, the *Starovic* reference states: "With an embodiment of the invention, it is not necessary to provide a separate level of control, for example a common operating system with shared storage, to ensure fault tolerance." [Emphasis added.]

As such, Applicant submits that it would not be obvious to the skilled person to combine *Osen* with another document such as *Starovic* that clearly teaches away from one of the important features of the present independent claims.

Applicant has also amended independent claims 1, 10, and 21 to incorporate the feature that write operations to the common heap are only performed by one of the first and second virtual machines. This feature corresponds in part to the feature specified in original claim 7. In support of the obviousness rejection, the Examiner refers to column 3, lines 20-37 of *Osen*, as well as column 8, lines 58-61 and column 9, lines 17-20 of *Starovic*, as allegedly rendering the above-mentioned feature obvious to a person having ordinary skill in the art.

In response, Applicant submits that the first and second jobs mentioned in column 3, lines 20-37 of *Osen* are clearly different from each other. The first job "scans a database and creates a long list of the physical addresses of all records, plus calculates a CRC-32 value from the addresses in said list as signature." On the other hand, the second job "does not save the physical addresses in the list, but only uses said addresses to compute the CRC-32." As

there are no virtual machines or other additional software layers in the system of *Osen*, it is clear that the difference in operation between the two jobs cannot be caused by a different execution environment, but must rather be coded into the jobs themselves.

Consequently, even if the ordinarily skilled person combined the teachings of *Osen* and *Starovic* in the manner proposed by the Examiner, the result would be a system that executes two or more different jobs in the respective virtual machines. This is clearly not what is claimed in the feature of claims 1, 10, and 21 that specifies that "the program is executed both by the first and by the second virtual machine." [Emphasis added.]

For the sake of clarification, Applicant notes that there may be other embodiments in *Osen* in which a single job is executed twice. However, these embodiments clearly do not comprise the feature that write operations to the common heap are only performed by one of the first and second virtual machines, as specified in the presently claimed subject matter.

On page 8 of the Office Action, the Examiner refers to column 8, lines 58-61 of *Starovic*, which states that the primary virtual machine resolves certain non-deterministic choices and informs the backups about its decisions. The Examiner infers from this statement that certain types of actions are only performed by the first virtual machine.

To avoid any possible confusion, it is submitted that, in *Starovic*, all actions resulting from non-deterministic choices will be performed by both virtual machines. For example, column 8, lines 54-55 of *Starovic* states that the non-deterministic choice may be the order of external interrupts. If there is more than one external interrupt request pending at a given time, then clearly the two or more virtual machines of *Starovic* must agree on a certain order of processing the interrupts so that consistency between the virtual machines is maintained. *Starovic* solves this problem by the primary replica choosing one order and informing the backup replica of the choice. However, all interrupt requests will still be processed by all replicas.

In the Office Action, the Examiner then refers to column 9, lines 17-20 of *Starovic* as allegedly disclosing that the "actions taken only by the first virtual machine include a write operation." Applicant respectfully submits that the above observation of the Examiner is not correct from a technical point of view. As indicated above, even if only the primary replica makes a non-deterministic choice (and informs the other replicas of the result), then the actions following the non-deterministic choice will still be processed by all replicas. These actions are exemplified in column 9, lines 17-20 of *Starovic* as reading from the environment, writing to the environment, and asynchronous actions from the environment. However, while only the primary replica may choose an order in which a number of pending write actions are performed, the actual write actions (in the determined order) will still be performed by all replicas. This is, in fact, clearly expressed in column 9, lines 17-20 of *Starovic*, which states that the actions mentioned above "require additional processing by the replicas [emphasis added] in order to resolve non-deterministic choices." If the actions were only performed by one replica, then there would be no "additional processing by the replicas," but, to the contrary, less processing by the replicas.

Yet further, no part of *Starovic* can be construed to teach that "write operations to the common heap are only performed by one of the first and second virtual machines," as presently claimed. As indicated above, the replica virtual machines of *Starovic* are located on different nodes of a computer network, wherein the nodes are linked by a network connection or an additional connection (column 9, lines 40-41) and do not have any shared storage (column 5, lines 46-48).

Finally, Applicant notes that the central purpose of *Starovic* is to create a software fault tolerant system that continues to operate even if one of the virtual machines fails. Column 9, lines 22-24 of *Starovic* states: "After a failure of the primary VM is detected, a backup VM replica is promoted to become the new primary VM replica." However, this

approach has the danger that the operation of the backup VM replica may have also been compromised by the event (such as a power spike or external manipulation) that led to the failure of the primary VM replica.

The teaching of *Starovic* thus squarely contradicts the teaching of the present invention, which is to abort execution of the program if a possible malfunction is detected. The presently claimed subject matter is therefore particularly tailored to security-critical applications such as financial transactions or the electronic signature of documents, where deliberate attacks by external interference must be countered (see Paragraph [0003] of the specification). Applicant submits that one having ordinary skill in the art would not use the teaching of *Starovic* when trying to develop a system that fulfills the security requirements of the present invention.

Thus, in view of the foregoing, the combination of *Osen* in view of *Starovic* would not have rendered the subject matter defined in present claims 1, 10, and 21 obvious to one having ordinary skill in the art. Accordingly, independent claims 1, 10, and 21, as amended herein, are patentable under 35 U.S.C. § 103(a) over the combination of *Osen* in view of *Starovic*.

Dependent Claims

The dependent claims are patentable at least by virtue of their dependency from patentable independent claims.

Furthermore, particular reference is made to dependent claims 8, 19, and 24, which recite that the second virtual machine, instead of performing the write operation to the common heap, checks whether a value that is to be written is present in the heap at the location that is to be written to.

The above-mentioned feature ensures detection of memory write errors in the memory in which the common heap is located (usually an EEPROM). This is an advantageous further

development of the present invention, considering the danger that an attacker may try to interfere with the memory writing process by means of voltage pulses, radiation, and so on (see Paragraph [0003] of the specification). *Osen* may teach to compare the results of two jobs, but there is no teaching in *Osen* that also detects if a result has properly been stored in memory.

Conclusion

In view of the foregoing, Applicant respectfully requests reconsideration and reexamination of claims 1-5, 7-10, 12-16, 18-21, and 23-25, as amended herein, and submits that these claims are in condition for allowance. Accordingly, a notice of allowance is respectfully requested. In the event a telephone conversation would expedite the prosecution of this application, the Examiner may reach the undersigned at **(408) 749-6902**. If any additional fees are due in connection with the filing of this paper, then the Commissioner is authorized to charge such fees to Deposit Account No. 50-0805 (Order No. WACHP008).

Respectfully submitted,
MARTINE PENILLA & GENCARELLA, L.L.P.

/Peter B. Martine/

Peter B. Martine
Reg. No. 32,043

710 Lakeway Drive, Suite 200
Sunnyvale, California 94085
Customer Number 25920